

Integrating DGSs and GATPs in an Adaptative and Collaborative Blended-Learning Web-Environment

Vanda Santos

ESTGV – IPV
Viseu, Portugal

CISUC, University of Coimbra
Coimbra, Portugal
vsantos@estv.ipv.pt

Pedro Quaresma

CISUC/Department of Mathematics
University of Coimbra
3001-454 Coimbra, Portugal
pedro@mat.uc.pt

The area of geometry with its very strong and appealing visual contents and its also strong and appealing connection between the visual content and its formal specification, is an area where computational tools can enhance, in a significant way, the learning environments.

The common experience is that dynamic geometry software systems (DGSs) significantly help students to acquire knowledge about geometric objects and, more generally, to acquire mathematical rigour. Using geometry automated theorem provers (GATPs) capable of construction validation and production of human readable proofs, will consolidate the knowledge acquired with the use of the DGSs. It will also help students to comprehend the connection between a concrete representation of a particular geometric construction and its formal description.

An adaptative and collaborative blended-learning Web-environment where the DGS and GATP features could be fully explored would be, in our opinion, a very rich and challenging learning environment for teachers and students.

In this text we will describe the Web Geometry Laboratory, a Web environment incorporating a DGS and a repository of geometric problems, that can be used in a synchronous and asynchronous fashion and with some adaptative and collaborative features.

As future work we want to enhance the adaptative and collaborative aspects of the environment and also to incorporate a GATP, constructing a learning environment for geometry, a dynamic system where the student has at his/her disposal the necessary tools for the study of theories and models of geometry. A system in which students can be challenged by new problems. A system that allows students the opportunity to collaboratively develop and improve their study in the area of geometry anywhere, at anytime and at their own pace.

1 Introduction

The use of computational tools in a learning environment can greatly enhance its dynamic, adaptative and collaborative features. It could also extend the learning environment from the classroom to outside of the fixed walls of the school. In the high-school curriculum in Portugal the use of such tools is praised. Quoting from the official curriculum specification¹ [4]:

The computer, by its own potential, namely in the areas of dynamic geometry, function representation and simulation, allows activities of exploration and research and also activities of recovery and development, in such a way that it constitutes an important asset to teachers and students, its use should be considered compulsory in this curriculum.

¹Translated from the Portuguese text.

The dynamic geometry software systems (DGSs) allow an easy construction of geometric figures built from free objects, elementary constructions and constructed objects. The dynamic nature of such tools allows its users to manipulate the positions of the free objects in such a way that the constructed objects are also changed, yet preserving the geometric properties of the construction. These manipulations are not formal proofs, the user is considering only a finite set of concrete positions, nevertheless these tools provide a first, not yet formal, link between the theories and models of geometry.

The Pythagoras theorem is a good example where a DGS can be used in a very fruitful way. The dynamic component allows us to work the visual proofs of this theorem. The JGEX Web page [1] has some very nice examples of visual proofs of this theorem.

The DGSs allow also to perform more complex geometric transformations like translations, reflections, rotations. The advantages of the DGSs in a learning environment are multiple: they are easy to use, they stimulate the creativity and the discovery process. There are multiple DGS available: GeoGebra, Cinderella, GeometerSketchpad, C.a.R., Cabri and GCLC [5, 6, 9, 12, 15, 21] to name some of the most used.

Automated theorem provers are less widely used as tools in a learning environment, but geometry with its axiomatic nature is a “natural” area for a formal tool such as the GATP. The GATPs give its users the possibility to reason about a given DGS construction, this is no longer a “proof by testing”, but an actual formal proof. If the GATP is capable of producing synthetic proofs, the proof itself can be an object of study, in other cases only the conclusion matters [2, 13]. Another link between the GATPs and the DGSs is given by the automated deductive testing, by the GATP, of the soundness of the constructions made by the DGS [10]. Most, if not all, DGSs are capable of detecting and report syntactic and semantic errors, but the verification of the soundness of the construction is beyond their capabilities. If we have this kind of integration between DGSs and GATPs we can check the soundness of a given construction. Again, if the GATP produces synthetic proofs, the proof itself can be an object of study, providing a logical explanation for the error in the construction. In either cases (formal proofs or soundness of the constructions) we claim that the GATPs can be used in the learning process [10].

To build an adaptive and collaborative blended-learning environment for geometry, we claim that we should integrate a DGS, a GATP, a repository of geometric problems (RGP), and the student model of interaction² in a Web system capable of asynchronous and synchronous interactions. A system with that level of integration will allow building an environment where each student can have a broad experimental, but with a strong formal support, learning platform.

Such an integration is still to be done, there are already many excellent DGSs, some of them have some sort of integration with GATPs, others with RGPs [13, 20]. Some attempts to integrate these tools in a learning management system (LMS) have already been done [23], but, as far as we know, all these integrations are only partial integrations. A learning environment where all these tools are integrated and can be used in a fruitful fashion does not exist yet.

In this text we describe the Web Geometry Laboratory (WGL) an asynchronous/synchronous Web environment that integrates a DGS program and a repository of geometric problems, aiming to provide an adaptive and collaborative blended-learning environment for geometry.

Paper overview. In section 2 we will describe the features needed in an adaptive and collaborative blended-learning Web-environment for geometry. In section 3 we present the Web Geometry Laboratory, a system that aims to be an environment of that type, we also describe the technical challenges that had

²These last two with the help of a database management system (DBMS)

to be tackled in its implementation. In section 4 we present a case study and in section 5 we speak about the work still to be done and we draw some final conclusions.

2 Goals for an Adaptative and Collaborative Blended-Learning Web-Environment for Geometry

If we want to build an adaptative and collaborative blended-learning environment for geometry what are the features and tools we are looking for? The following list is, in our opinion, close to a complete, minimal, set of the features and tools needed for building such a system.

2.1 Geometric Tools

As said above, the advantages of the DGSs in a learning environment are multiple: they are easy to use, they stimulate the creativity and the discovery process. They provide an outstanding tool to substitute the old ruler and compass used in the classrooms. The constructions made from free objects and constructed objects allow a degree of property preserving manipulations much superior to the capabilities of physical tools.

In spite of the DGSs outstanding features, they do not create a learning environment by themselves, so its integration in a learning environment will be beneficial. The DGSs excel in the dynamic construction of geometric figures, the learning environment should add to this, the collaborative and adaptative features.

The DGSs provide a first, not yet formal, link between the theories and models of geometry. But if we want to reason about the constructions we are doing, to make conjectures about their proprieties or in a more generic way to make formal deductive reasoning about geometric constructions, we need more than a DGS, we need to extend the reasoning from concrete instances in a given model to formal deductive reasoning in a geometric theory. To have this we need to add to our environment a GATP capable of synthetic proofs (e.g. the Area Method [11]) or algebraic proofs (e.g. the Wu's Method [24] or the Gröbner Basis Method [14]).

A GATP will be helpful, at least, at two distinct tasks:

- Construction validation: most (if not all) DGSs use only geometric concepts interpreted from concrete instances in the Cartesian plane. A construction is always associated with a concrete fixed set of points (with concrete Cartesian coordinates). In such environments, some constructions are illegal (e.g., if they attempt to make the intersection of parallel lines), but the question if such construction is always illegal or it is illegal only for a given particular set of fixed points is left open. Indeed, for answering such questions, one has to use deductive reasoning, and not only a semantic check for that special case. On those situations using a GATP we could get the information that the construction is illegal, and moreover, that it is illegal not only for a given special case, but always. In this way, the deductive nature of geometrical conjectures and proofs are linked to the semantic nature of models of geometry and, also, to human intuition and to geometric visualisations [10].
- Geometric proofs: proofs are exemplary mathematical contents. They can serve in mathematical education, aimed at acquiring mathematical rigour.

Using a synthetic proof style GATP, e.g. an area method based GATP [11], we may have access to the proof itself. In these cases the proof itself will be an object of study.

An integration between a DGS and a GATP like the integration that can be found in the GCLC system [12, 13] provides an environment where a user can reason about geometric conjectures, learning about the links between a formal theory and its models.

The connection with a repository of geometric problems (RGPs) will add to the environment, memory, i.e., the capability to save/recover geometric constructions. The teacher would be able to prepare in advance a set of constructions/exercises to be released to the students. The students would be able to keep their constructions in a personal *scrapbook*.

With a connection to a database management system (DBMS) we could foresee the possibility of having a repository of geometric constructions and/or individualised scrapbooks, but also the possibility of constructing the students' model, keeping the history of the students' interaction with the system, in this way allowing the adaptation of the environment to each student [7].

2.2 Blended-Environment

A learning environment for geometry should be, in our opinion, a blended-learning environment³. It should be an environment that can be used as a geometry laboratory in a classroom by teachers and students in a much enhanced substitute for the ruler and compass physical instruments. But it should also allow to extend itself outside the classroom, for homework tasks, for problems proposed by the teacher to be solved outside the classroom, for students tutored study, where the tutors could be the teacher(s) and/or classmates.

A Web-environment is appropriate for both situations, in a classroom a local or wide area network (LAN or WAN) environment would allow the synchronous interaction between teacher and students. Outside the classroom a WAN environment would allow synchronous and asynchronous interactions. The ubiquity of the Web nowadays make such a learning environment easily accessible.

2.3 Collaborative Environment

The environment should be collaborative, i.e. it should allow the knowledge to emerge and appear through interaction between its users [8]. The teacher and students should be able to interact with each other but the environment should also be open to interaction between students. Speaking about a geometric environment this interaction should not be restricted to textual contents, (chats, wikis, etc.) but it should be extended to geometric contents. i.e. the users of such a system should be able to exchange geometric constructions, or even to build a geometric construction in a collaborative way. This exchange of geometric contents will open new possibilities in terms of a collaborative geometric learning environment.

2.4 Adaptative Environment

The environment should be adaptative, i.e. it should adapt the help information given to different users and also, an important feature in a learning environment, to adapt the learning path to the different users needs [7, 16, 17].

The system should be able to infer the geometric knowledge of the users or to use plan recognition, in terms of geometric knowledge, to infer the actual plan or information goal of the users [7]. We see this as an important challenge to be addressed by any new system.

³A blended-learning environment is a mixing of different learning environments. It combines traditional face-to-face classroom (synchronous) methods with more modern computer-mediated (asynchronous) activities.

Defining a network environment that integrates a DGS, a GATP and a DBMS (for the RGP and for the students' model) will provide more than the simple sum of its components. It will allow the construction of an adaptative and collaborative blended-learning environment for geometry.

3 Web Geometry Laboratory

The *Web Geometry Laboratory*⁴ (WGL) is a Web environment (see Figures 1 and 2), that integrates a DGS program and a repository of geometric problems (RGP), aiming to provide an adaptative and collaborative blended-learning environment for geometry.

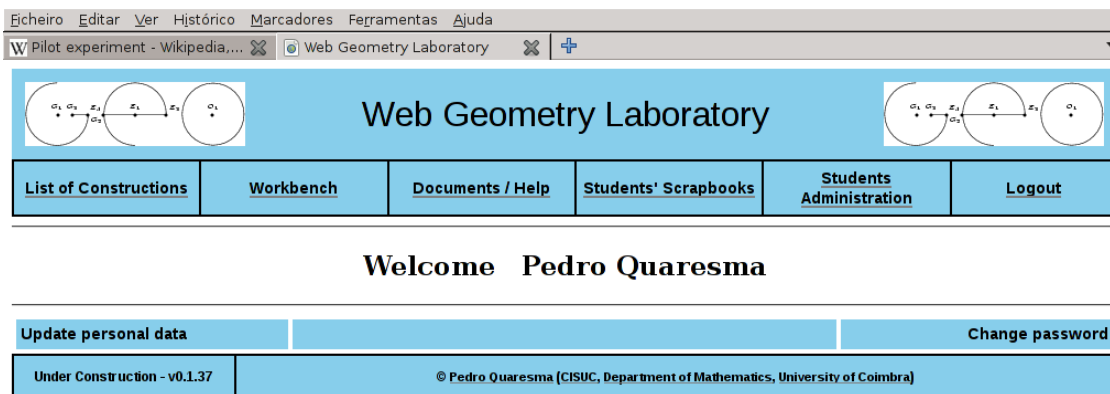


Figure 1: Web Geometry Laboratory — Teacher's Login

The WGL system allows the teacher to create, store and provide a set of geometric constructions to its students. The WGL system allows the student to access the professor's constructions as well as those kept in a personal *scrapbook*. The students' scrapbook (see Figure 2) is a place where the student will keep his/her own constructions, solutions of problems placed by the teacher and/or his/her own exploratory activities. The teacher has also access to the constructions made, or being made, by the students as a way to be able to help the student during a class or to evaluate the work done after class, or even as a mean to broadcast the work done by a student to the rest of the class. In a next version of WGL the students will be able to work collaboratively, seeing and exchanging each other constructions.

The WGL aims to provide a learning environment for geometry using all the potential of a given DGS, all the easiness of access provided by an Web platform and with an individualised memory provided by a database where all the history of each student is kept. With the inclusion of a GATP and the implementation of some, now missing, features, the WGL aims to become an adaptative and collaborative blended-learning environment for geometry.

From the point of view of the server the system aims to be easy to install, maintain and use. From the point of view of the clients, teachers and students, the only feature that is needed is the access to a Java aware⁵ Web browser, which is, nowadays, a minor issue given the ubiquity of such type of program.

If installed in the school server the WGL will not be confined to the classroom, it can be opened to the global network, thus enabling its remote use by teachers and students, extending its use to a blended-learning environment. The WGL will allow teachers to set new challenges to be solved outside the

⁴A test site is available at <http://hilbert.mat.uc.pt/WebGeometryLab/>

⁵The WGL uses the GeoGebra Applet

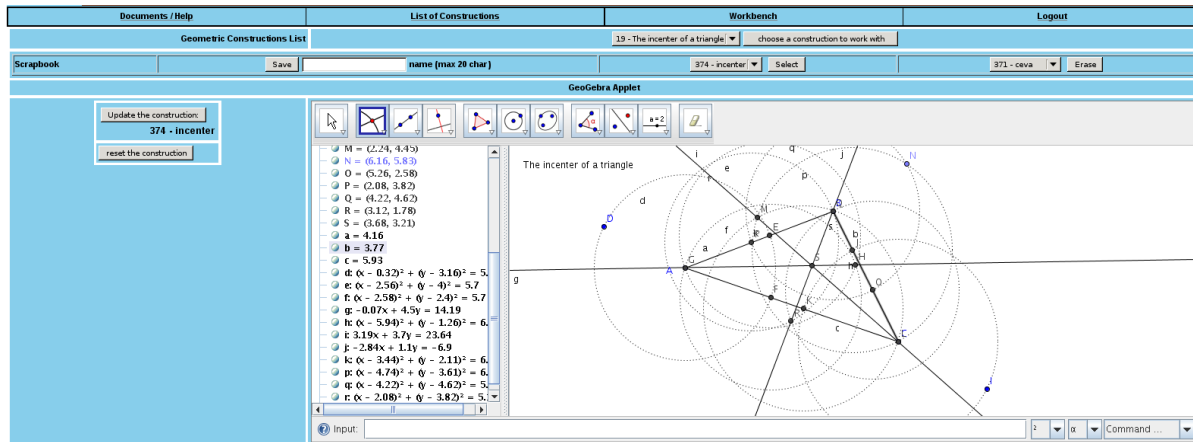


Figure 2: Web Geometry Laboratory — Workbench

classroom, allowing each individual student to work at his/her own pace.

Defining a network environment that integrates a DGS, a GATP and a database, the *Web Geometry Laboratory* will provide more than the simple sum of its components. Integrating the DGS and the GATP fully in a Web-environment will open the use of the DGS and GATP to a collaborative blended-learning environment. Linking all that with a database will allow an individualisation of the learning environment with the creation of an adaptive (individualised) learning environment.

3.1 Levels of Access

The access and privileges of the WGL mimics the Unix access and privileges system. Any user of the WGL will belong to one of the three possible levels of access, giving him/her a different initial Web-page and a different set of privileges over the geometric constructions on the repository.

As said above, the access to WGL has an hierarchy with three levels of access: administrators, teachers and students.

- the *administrator's level* that allows the system administrator to define whose users will have *teachers status* within the system;
- the *teachers' level* (notice the menu bar in Figure 1), which allows access (with validation) for teachers. Through this module it will be possible to teachers to make the management of students. It provides also access to the workbench in order to build and set the constructions that will be accessible to all the students, and finally it gives access to the students' *scrapbooks*;
- the *students' level* provides access, through a validation process, for students (see Figure 2, notice the menu options). This module allows access, read-only, to the constructions provided by the teachers. Full access to the student's personal *scrapbook*. Last (but not least) each student will be able to open his/her working area (read and write access) to others in order to be able to work collaboratively.

In the actual version of WGL, apart from the definition of the different levels of access, the definition of the privileges has a very naive implementation. Any construction in the repository has a "level attribute". Negative numbers in this attribute means that the construction is only available to teachers, a positive number means it is available for all (See Figure 4). This naive solution will be changed to a

Unix-like permissions system, *users*, *groups* and *others* will have *read*, *write*, and *visibility* privileges. Every construction will be associated to a user, who will have all the privileges over the construction. Teachers will be capable of defining groups, any subset of the class' students, and to define which kind of privileges (read/write/visibility) the members of the group will have among them.

3.2 Technical Issues

The WGL is a multi-platform, multi-language, multi-tool, synchronous and asynchronous Web-environment. Some of these issues are not new and also not difficult to solve, others, like the smoothly integration of tools, raise some important and difficult technical questions.

In this section we will try to list all the questions that should be addressed by the programmer of such a system and the way the WGL had addressed, or will address, those questions:

- multi-platform: this is, in our opinion, a very important feature of an educational software because it opens its use to a wider audience. To build a multi-platform software it is important to choose the right tools and technologies. The WGL uses: *Apache* (Web server); *PHP* (Web script, server side, language); *MySQL* (DBMS); *AJAX* (asynchronous Javascript and XML); *JavaScript* (Web script, client side, language); *Java Applets* (Web language). All these tools and programming languages are multi-platform;
- multi-language: another important issue whenever we want to build a software system for the global world. In WGL the *gettext* library [3] for *PHP* is used. The *base* system is generic, having as default language the English. Adding to the base system a set of translations will provide localisation to the system. Currently only the Portuguese translation is provided;
- Web-environment: this issue is an easy to solve issue. From the server side the *Apache* Web-server is being used, but any other Web-server should be able to support WGL. The pages are being built using CSS/HTML/PHP (plus other technologies already mentioned). From the client-side any Java-aware browser will be able to run the environment;
- synchronous/asynchronous interaction: this is important at two different levels: the interaction of the user with the system and with each other; the interaction between tools. The first issue is a design question, the WGL blended-learning environment allows both a synchronous interaction (classroom) and an asynchronous interaction. The other question is very important in a multi-tool Web environment if we want to provide a fluid, easy to use, system where the interaction of the user with the DGS (synchronously) and the repository of problems (asynchronously) should be transparent to the user. That is, the user should be able to fetch a new construction and to load it into the DGS or, on the opposite direction, save the DGS construction into the database, and all this without glitches in the environment.

In WGL this question is solved using the AJAX asynchronous features, allowing loading/saving construction from/to the database without interfering with the DGS;

- integration of tools: this is the most difficult technical issue to be dealt with when building a system like WGL. Two main issues must be dealt with and solved: the communication between tools and its integration in terms of a fluid interaction with the users. Both step should be transparent to the users.

The integration of the DGS (GeoGebra) is done via the DGS API, the asynchronous update of the Web-page is done with the help of an AJAX call. The DGS API provides the construction in a XML format and this text is kept, and recovered when needed, in the database.

The integration of a GATP is still to be done. We are working on the I2GATP [18, 19], an extension of the I2G format [22] to cope with geometric conjectures and its proofs. The integration of an GATP, e.g. the *GCLCprover* [13], will use this format to communicate, via an API, with the DGS and the database.

A component still missing is the construction of the student's interaction model, we foresee no technical difficulties with this task given the fact that the asynchronous connection with the database is already done. In a similar way the collaborative features will be implemented using the asynchronous connections, allowing the exchange of geometric contents.

3.3 Installation

That system will be easy to install on a school server (freeing the teachers of technical issues), or even on the teachers' personal computers (given more liberty of use to the teacher). It will require:

- a computer hosting a network server (local or global) capable of providing access to the Web pages of the WGL, and a DBMS to provide access to the repository of problems and to the students' individualised memory. Any Linux/MacOS computer will be capable of providing such an environment, it will also be possible to build such an environment on top of a MS-Windows system;
- access to a network (local or global). The school network or even, with the help of an wireless access point, a local/classroom wireless network provided by the teacher;
- computers containing a Web browser connected to the network for teachers and students access to the system. Nowadays any computer is equipped with such a program, if not, installing one is always possible.

All this can be built using open source programs so, apart from the hardware, the WGL system has no other costs, and even the hardware necessary to support the server and the clients will not represent a great burden.

4 A Case Study

We will now try to do a *walking tour* throughout WGL by means of an example. We will try to describe the possibilities of the current implementation and also to point out some of the improvements already thought out.

For this example we will assume that all the registration of teachers (by an administrator of WGL) and students (by their teachers) is already done.

The example will be about the notable points of a triangle, e.g. the incenter, the circumcenter and the orthocenter. The lesson should have two distinct moments: during the class the introduction of the incenter and its construction by the students using the WGL; after the class the other two notable points.

Before the class, during the planning phase, the teacher will use the WGL to prepare a set of constructions to be delivered during the class: an initial construction with the triangle (Figure 3); the construction of the first internal angle bisector (Figure 5); the intersection of the three angle bisectors, the incenter, (Figure 6); and the incircle (Figure 7).

For every one of these constructions the teacher can decide which ones will be visible to the students and which ones will be only visible to himself/herself. In figure 4 we can see the teacher's and the students' view of a given constructions list.

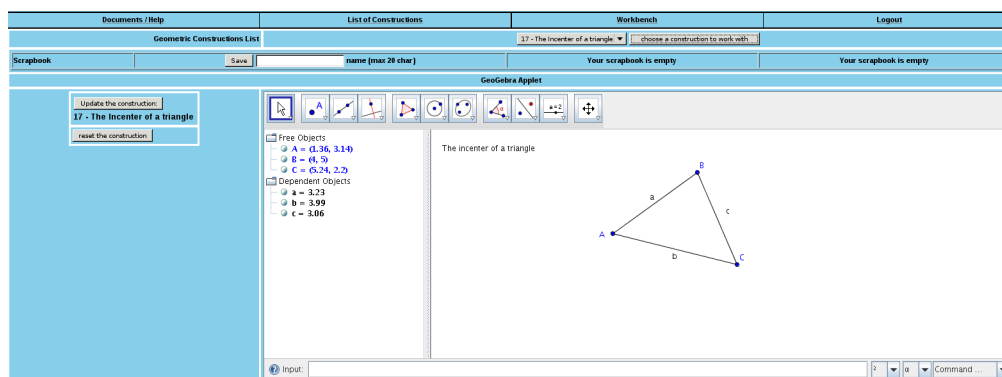


Figure 3: The Initial Construction

Name		Add a new construction			
The Incenter of a triangle	The incenter of a triangle is the center of	4	See Details	Update	Delete
Teorema de Ceva	Teorema de Ceva	-32	See Details	Update	Delete
Teorema de Pappus	Sejam A, B e C três pontos numa recta r, CB1 com C1B. Prova-se que os pontos P, C1 com A1C e R a intersecção de	-32	See Details	Update	Delete

Figure 4: Constructions List / Access Privileges

At a certain point during the class, maybe after an initial introduction of the notable points of a triangle, the teacher will ask the students to login into the WGL, load the initial construction (see Figures 3 and 4), and to begin work on the first task, the construction of the angle bisector (see Figure 5).

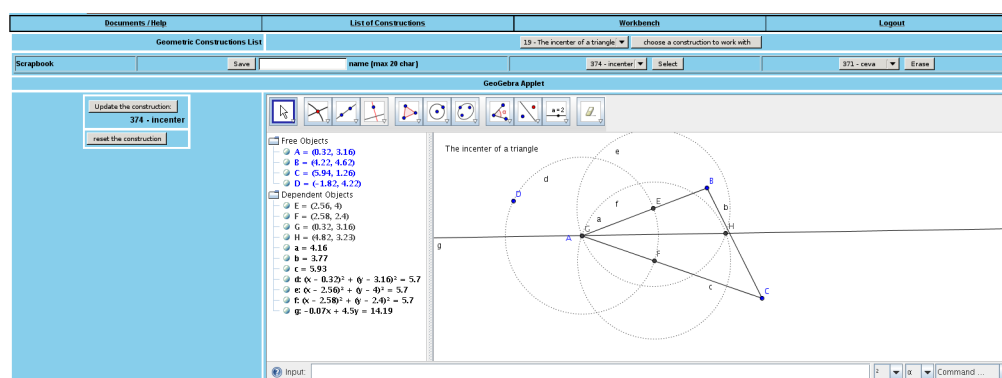


Figure 5: Angle Bisector

At this point all the students can access the construction made by the teacher and they can begin working individually. In a collaborative perspective the students should be allowed to exchange their work among the groups and even to be able to make the construction collaboratively. In the current implementation of WGL only the teacher has access to all the students work (notice the different tabs

with the students name in Figure 6). This should be greatly improved in a next version of WGL, the implementation of the users' privileges system should allow extending this to groups of students working collaboratively.

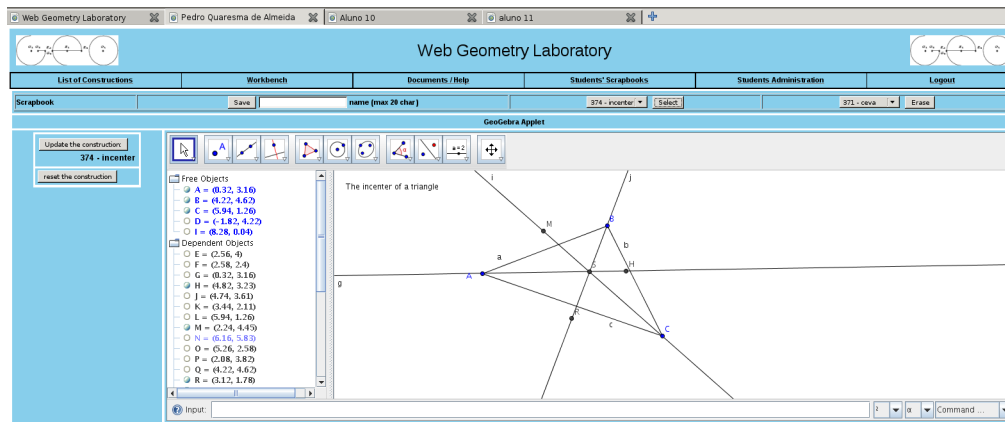


Figure 6: Incenter & Students' Tabs

As said above the teacher has access to all the students' constructions. At any moment he/she can come to help a student or a group of students and, when the task is complete, he/she can broadcast the solution, as the starting point of a new task, e.g. finding the other bisectors and their intersection point (see Figure 6). The solution broadcast by the teachers can be a previously prepared (by the teacher) construction or one of the students' solutions.

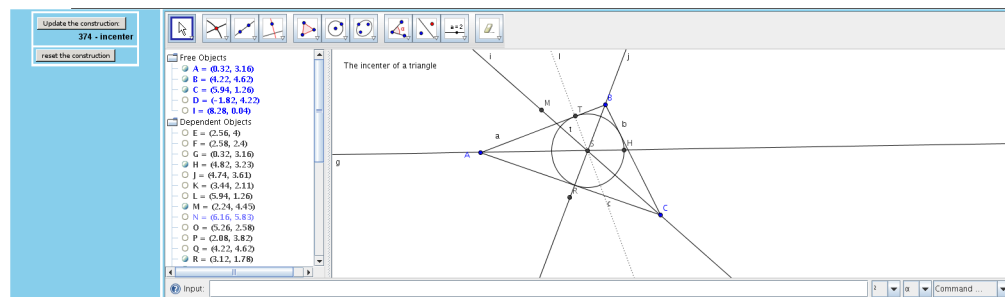


Figure 7: Incircle

The teacher should always stress out the constructive nature of the geometric figure being made. From a set of free points and using a well defined set of constructive steps, another set of points is being constructed. Having finished the incenter construction, the dynamic possibilities of the DGS could be used to show that this construction is more generic that it seems at a first glance. This could, and should be, used to open the door to the rigorous, or even formal, proof of this result. When the connection between WGL and a GATP (synthetic or algebraic) will be completed, it will be possible to write down a geometric conjecture, related to the construction being made, and then, calling an appropriate GATP, to have access to its proof, or at least to the proof result.

At the end of the class the teacher will introduce the other notable points of the triangle, e.g. the circumcenter and the orthocenter, making visible their initial constructions and setting the tasks to be performed by the students, with the help of WGL, but outside the classroom. Again using the users'

privileges system this could be done in a collaborative setting.

All the constructions made by the students, are kept in their personal scrapbook. This is not, yet, an adaptative system, but it gives already a certain degree of individualisation, giving to each student a track record of his/her own work. The teacher has read access to all the students' scrapbooks. This can be used to monitor their work and/or to be used to grade the students, evaluating all the work done by each student, during the classes and also outside the classes.

5 Conclusions & Future Work

With the integration of these three tools: a DGS, a GATP and a DBMS we aim to build an adaptative and collaborative blended-learning environment for geometry.

Conclusions. The integration of dynamic geometry tools in the school environment allows to diversify the study of geometry, enhancing its dynamic side. One objective is to place the students in front of a “visual demonstration” of several case studies. The ability to perceive different representations of the same construction is a strategic point for the students development. The control of geometric configurations leads to the discovery of new and interesting properties. The learning process allows to make experiences, develop strategies, make conjectures, reason and deduce mathematical properties, and from this to begin using the geometric automated theorem provers to introduce formal deductive reasoning.

The *Web Geometry Laboratory* will allow teachers to use the DGSs and GATPs in a more fruitful way. The teacher will be able to prepare a set of constructions in advance and to provide them to an entire class easily. It will also allow the teachers to monitor each student, during the class and after the classes. At the same time students will have an individual platform for learning geometry at their own pace.

The goal of this project is to build a dynamic adaptative and collaborative blended-learning environment for geometry, incorporating a dynamic geometry software tool, one, or more, geometric automated theorem provers and a repository of geometric problems, in an environment where the student has the necessary tools at his/her disposal for the study of theories and models of geometry. A system where the student can understand the differences and connections between these two perspectives and improve their knowledge. A system in which the student can also be challenged by new problems, giving the student the opportunity to develop and improve their study in the area of geometry anywhere, at any time and at their own pace.

Future Work. The *Web Geometry Laboratory* is a “work in progress” project. As a first task we need to complete a first prototype of a standalone system capable of being distributed to schools and/or teachers. Such a system should already include a set of geometric constructions and a course syllabus to help teachers to organise their work. The integration of a GATP and/or an interactive theorem proving (ITP) is also a wanted feature and it is planned as a task to be pursued in parallel with the described tasks.

The adaptative and collaborative features should be studied to have a better understanding of how to use them in a geometric context.

References

- [1] Shang-Ching Chou, Xiao-Shan Gao & Zheng Ye (2004): *Java Geometry Expert*. <http://www.cs.wichita.edu/~ye/>.

- [2] Shang-Ching Chou, Xiao-Shan Gao & Jing-Zhong Zhang (1996): *Automated Generation of Readable Proofs with Geometric Invariants, I. Multiple and Shortest Proof Generation*. *Journal of Automated Reasoning* 17, pp. 325–347, doi:10.1007/BF00283134.
- [3] Ulrich Drepper, Jim Meyering, François Pinard & Bruno Haible (2010): *GNU gettext tools, version 0.18*, 0.18 edition. Free Software Foundation, Inc.
- [4] Departamento do Ensino Secundário (2011): *Matemática A, Cursos Gerais de Ciências Naturais, Ciências e Tecnologias, Ciências Sócio-Económicas*. Technical Report, Ministério da Educação, República Portuguesa.
- [5] René Grothmann (2011): *About C.a.R.* <http://compute.ku-eichstaett.de/MGF/wikis/caruser/doku.php?id=history>.
- [6] M Hohenwarter (2002): *GeoGebra - a software system for dynamic geometry and algebra in the plane*. Master's thesis, University of Salzburg, Austria.
- [7] Dorothea Iglezakis (2004): *Adaptive Help for Webbased Applications*. In: *Adaptive Hypermedia and Adaptive Web-based systems*, LNCS 3137, Springer-Verlag, pp. 304–307, doi:10.1007/978-3-540-27780-4_38.
- [8] Tomoo Inoue & Naoko Yoshimura (2010): *EduWiki: A knowledge Construction System for Collaborative Learning*, chapter 11, pp. 167–176. *Advances in Learning Processes*, I-Tech Education and Publishing KG, Vienna, Austria.
- [9] N Jackiw (2001): *The Geometer's Sketchpad v4.0*. Key Curriculum Press.
- [10] Predrag Janičić & Pedro Quaresma (2007): *Automatic Verification of Regular Constructions in Dynamic Geometry Systems*. In: *Automated Deduction in Geometry*, LNAI 4869, Springer, Berlin / Heidelberg, pp. 39–51, doi:10.1007/978-3-540-77356-6_3. 6th International Workshop, ADG 2006, Pontevedra, Spain, August 31–September 2, 2006. Revised Papers.
- [11] P. Janičić, J. Narboux & P. Quaresma (2011): *The Area Method: a Recapitulation*. *Journal of Automated Reasoning* (to appear), doi:10.1007/s10817-010-9209-7.
- [12] Predrag Janičić (2006): *GCLC — A Tool for Constructive Euclidean Geometry and More Than That*. *Lecture Notes in Computer Science* 4151, pp. 58–73, doi:10.1007/11832225_6.
- [13] Predrag Janičić & Pedro Quaresma (2006): *System Description: GCLCprover + GeoThms*. In Ulrich Furbach & Natarajan Shankar, editors: *Automated Reasoning*, LNAI 4130, Springer, pp. 145–150, doi:10.1007/11814771_13.
- [14] Deepak Kapur (1986): *Using Gröbner bases to reason about geometry problems*. *Journal of Symbolic Computation* 2(4), pp. 399–408, doi:10.1016/S0747-7171(86)80007-4.
- [15] J. M. Laborde & R. Strässer (1990): *Cabri-Géomètre: A microworld of geometry guided discovery learning*. *International reviews on mathematical education- Zentralblatt fuer didaktik der mathematik* 90(5), pp. 171–177.
- [16] Miguel Angel Mora & Roberto Moriyón (2001): *Collaborative analysis and tutoring: the FACT framework*. In: *Advanced Learning Technologies, 2001. Proceedings. IEEE International Conference on*, pp. 82–85, doi:10.1109/ICALT.2001.943861.
- [17] R Moriyón, F Saiz & M Mora (2008): *GeoThink: An Environment for Guided Collaborative Learning of Geometry*, pp. 198–206. *Nuevas Ideas en Informática Educativa* 4, J. Sánchez (ed), Santiago de Chile.
- [18] P. Quaresma, Tomašević J. Janičić, P., M. V.-Janičić & D. Tošić (2008): *Communicating Mathematics in The Digital Era*, chapter XML-Bases Format for Descriptions of Geometric Constructions and Proofs, pp. 183–197. A. K. Peters, Ltd.
- [19] Pedro Quaresma (2011): *Thousands of Geometric problems for geometric Theorem Provers (TGTP)*. In P. Schreck, J. Narboux & J. Richter-Gebert, editors: *ADG 2010*, LNAI 6877, Springer, Heidelberg, pp. 168–180, doi:10.1007/978-3-642-25070-5_10.
- [20] Pedro Quaresma & Predrag Janičić (2006): *Integrating Dynamic Geometry Software, Deduction Systems, and Theorem Repositories*. In Jonathan M. Borwein & William M. Farmer, editors: *Mathe-*

- mathematical Knowledge Management, Lecture Notes in Artificial Intelligence* 4108, Springer, pp. 280–294, doi:10.1007/11812289_22.
- [21] Jürgen Richter-Gebert & Ulrich Kortenkamp (1999): *The Interactive Geometry Software Cinderella*. Springer.
- [22] E. Santiago, Maxim Hendriks, Yves Kreis, Ulrich Kortenkamp & Daniel Marquès (2010): *I2G Common File Format Final Version*. Technical Report D3.10, The Intergeo Consortium. Available at <http://i2geo.net/xwiki/bin/view/I2GFormat/>.
- [23] Vanda Santos & Pedro Quaresma (2008): *eLearning Course for Euclidean Geometry*. In: *Proceedings of the 8th IEEE International Conference on Advanced Learning Technologies, July 1st- July 5th, 2008, Santander, Cantabria, Spain*, pp. 387–388, doi:10.1109/ICALT.2008.156.
- [24] W.-T. Wu (2000): *The characteristic set method and its application*. In X.-S. Gao & D. Wang, editors: *Mathematics Mechanization and Applications*, Academic Press, San Diego, CA, pp. 3–41.